# Particle Identification using graphical models

Kevin Yarritu
LANL

10/31/2013

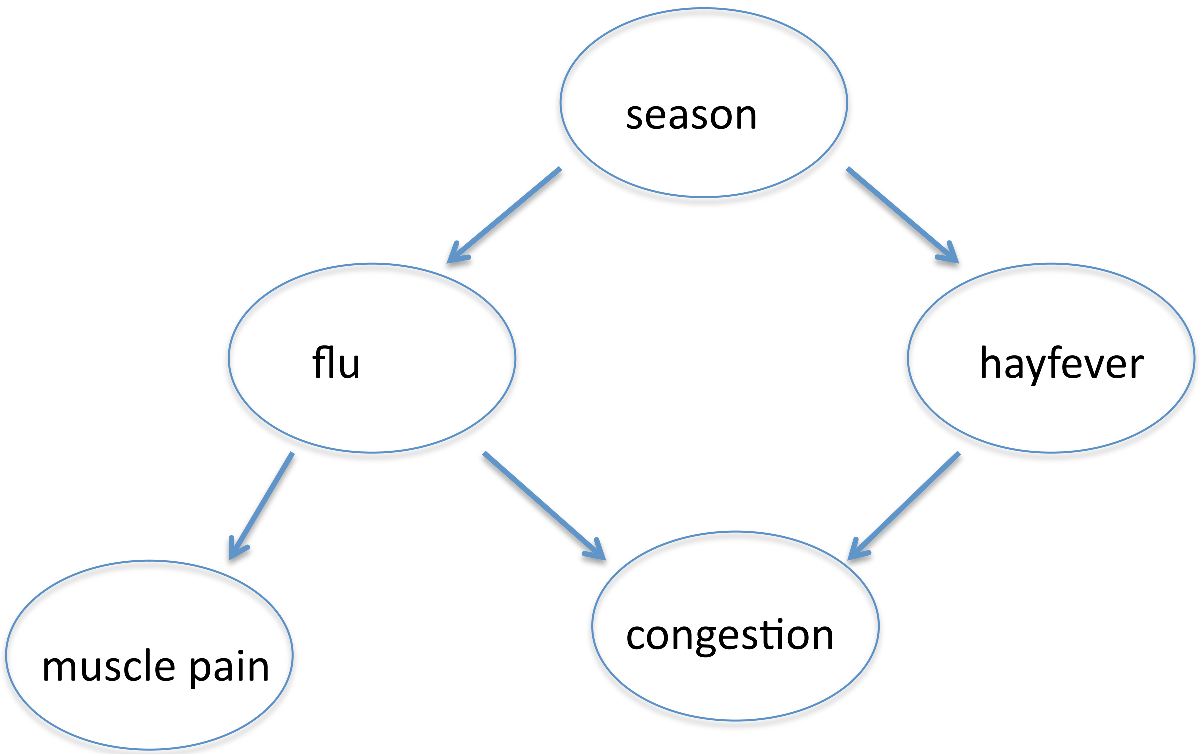
# Graphical models

- Probabilistic graphical models are a graph based representation for encoding a complex distribution over many random variables
- Used in a wide range of domains including computer vision, natural language processing, web search, medical diagnosis, and physics
- uses concepts such as Bayesian networks and Markov fields
- kind of like a Feynman diagram in the sense that it is a visual representation of a mathematical formalism

# Graphical models

- A Bayesian network is a probabilistic graphical model that represents a set of random variables and their conditional dependencies via a directed acyclic graph
- A Markov field is similar to a Bayesian network, but is undirected

# An example of a Bayesian network



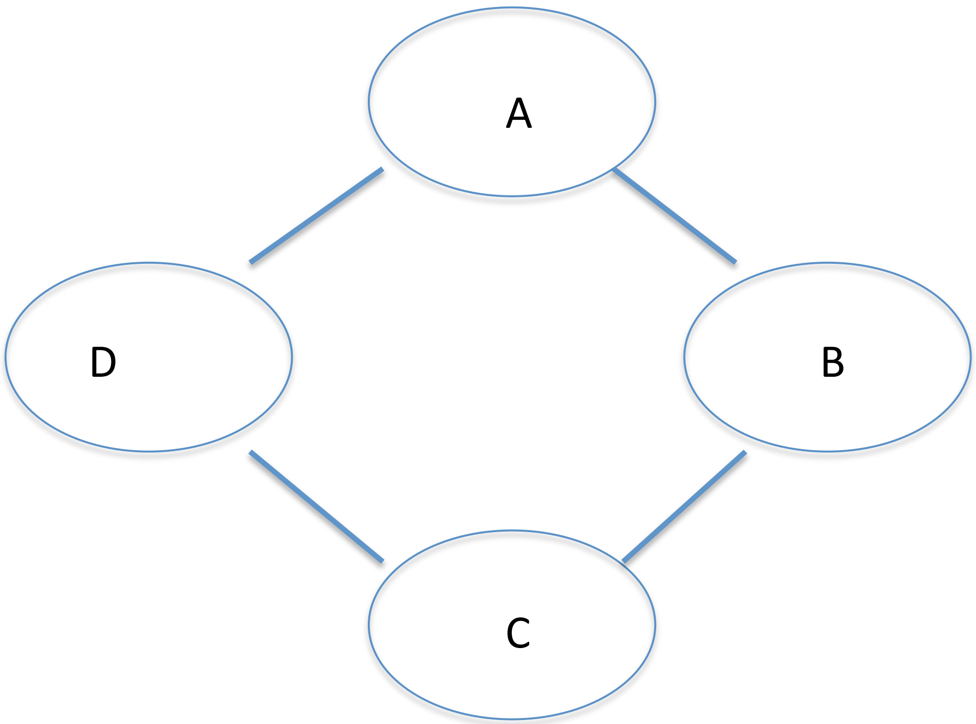

Independencies:
(F ,H | S)
(C,S | (F and H)
(M, (H and C) | F)
( M, C | F)

From the chain rule of probabilities:

P(S,F,H,C,M) = P(S) x P(F|S) x P(H|S) x P(C|F,H) x P(M|F)

- nodes represent random variables
- arrows represent dependencies among variables

# An example of a Markov network

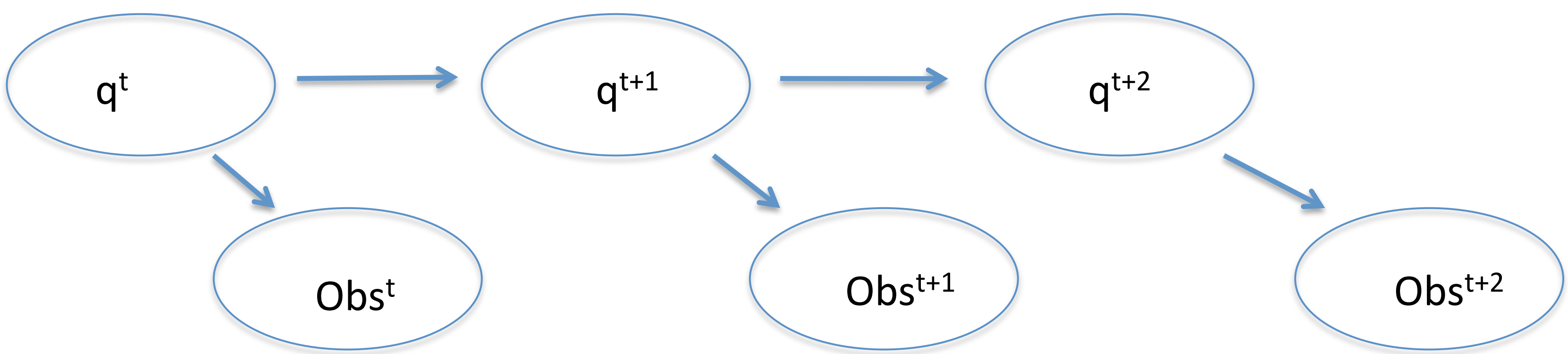



Independencies:
(A ,C | (B and D)
(B,D | (A and C)

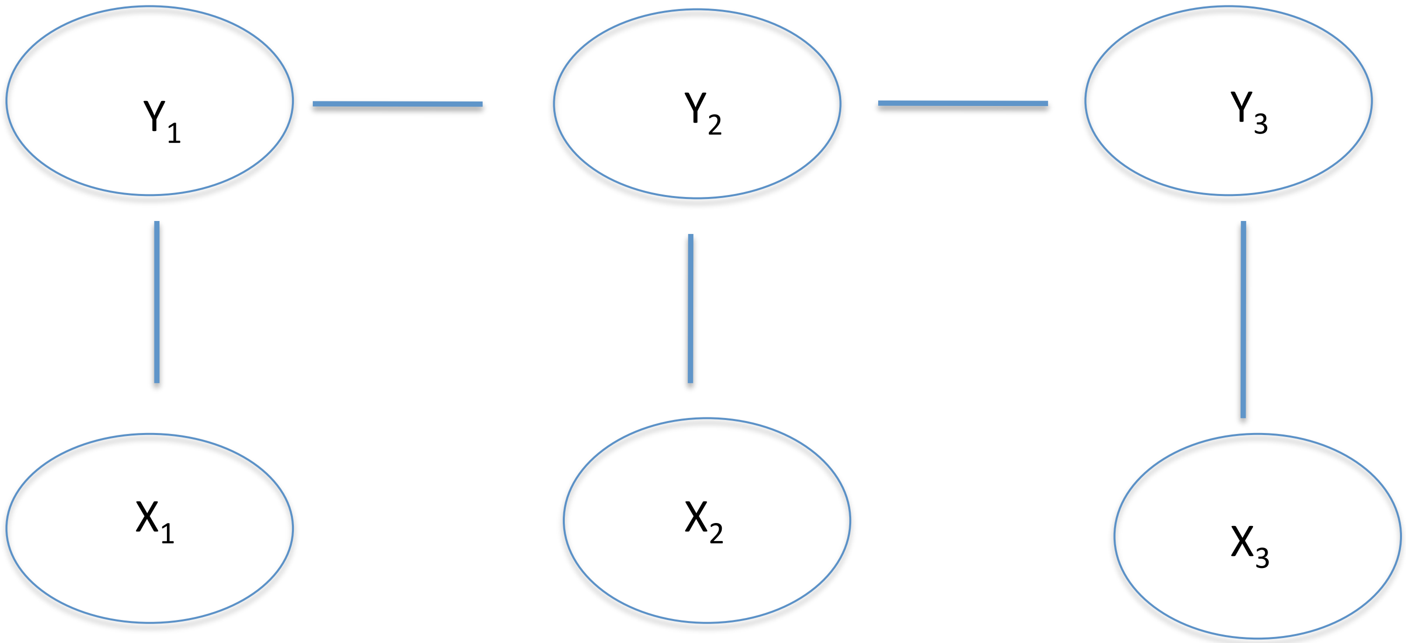
- P(A,B,C,D)= (1/Z) x $\Phi_1$(A,B) x $\Phi_2$(B,C) x $\Phi_3$(C,D) x $\Phi_4$(A,D)
- Z is normalization factor
- Factor ($\Phi$) represents the affinity between two variables

# Kalman filter

- Can be represented as a Bayesian network (called a dynamic Bayesian network) where the sate vector (q) is a random variable
- Proceeds through update equations
- $P(q^{t+1}|q^{t})$ represents the probability that the state will transition to $q^{t+1}$ given $q^{t}$. $P(Obs^{t}|q^{t})$ represents the probability of make an observation given a state $q^{t}$.
- simplest Kalman filter assumes state vector is Gaussian distributed



# Conditional random field




- y is a label, x is a feature

# Conditional random field

- Used for multivariate classification
- takes a set of input random variables and outputs a conditional probability distribution
- this is represented as exponential distribution

$$p(y|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \exp\left\{\sum_{k=1}^{K} \theta_k f_k(y, \mathbf{x})\right\}$$

- y is the class label, x is the feature
- Z is a normalizing constant
- f is a feature function that has to be used as input
- θ is the weight vector

# Conditional random field

- idea is to use this for particle identification
- take a set of input variables and output the conditional probability of a particular particle given those input variables
- these variables are undecided (dE/dx, range, shower distribution)

# Neutral network for electron-hadron discrimination

- Electron-Hadron shower discrimination in a liquid argon time projection chamber (arXiv.1210.2215)
- used 4 variables to discriminate between electrons and hadron
- 2 variables had to do with how the energy is distributed in the shower, and 2 were spatial extent

| MVA method | Beam case study | Electron–Signal/Background efficiency [%] | Signal-to-total $S/\sqrt{(S+B)}$ |
|---|---|---|---|
| KNN | CN2PY | 95.7 / 5.2 | 30.1 |
| MLPBNN | CN2PY | 93.9 / 7.4 | 29.5 |
| BDT | CN2PY | 96.2 / 4.5 | 30.3 |
| BDTG | CN2PY | 90.6 / 6.0 | 29.2 |
| KNN | Uniform $\nu$ | 95.4 / 5.0 | 30.1 |
| MLPBNN | Uniform $\nu$ | 92.9 / 7.3 | 29.4 |
| BDT | Uniform $\nu$ | 96.1 / 4.9 | 30.2 |
| BDTG | Uniform $\nu$ | 90.6 / 4.2 | 29.4 |

# conditional random field vs. neural network

- neural networks use a hidden layer, conditional random fields (mostly) do not use a hidden layer
- when learning the parameters, introducing a hidden layer loses convexity
- other methods should be attempted

# Conclusions

- conditional random fields offer a method of multivariate classification
- feature functions need to be determined
- input variables need to be defined